Department of Computer Science
Faculty of Science, University of Jaffna

GENERAL DEGREE COURSES IN COMPUTER SCIENCE

First Year Courses

Unit No.	Unit Title
CSG 11	Introduction to Computing and Pascal Programming
CSG 12	Theoretical Foundations of Computer Science
CSG 13	Computer Architecture and Assembly Language Programming
CSG 14	Data Structures

Notes:

1. Each of the Courses except CSG12 will consists of 35 hours lectures plus tutorials and 25 hours of practical
2. All but CSG 12 will be examined by a theory paper plus a practical examination
3. CSG12 will consists of about 45 hours of lectures plus tutorials and it will be examined solely by a theory paper.
4. A theory paper is of two –hour duration containing six questions out of which four should be answered.
5. A Practical examination is of three-hour duration.
6. Twenty Percent (20%) Weightage will be given for a Practical examination and eighty Percent (80%) Weightage will be given for a theory Paper. In case of CSG 102, one hundred percent (100%) is given for the theory paper.

CSG 11 : Introduction to Computing and Pascal Programming

Introduction to Computing: Functional parts of computer, Representation of information in a of computers, Concept of Programming and Programming languages, Input and output devices, An overview of computer Applications, Classification of computers and Computer Systems, Concepts of Algorithms.

Pascal Programming: Basic concept of data and data types, standard types in Pascal, Type statement, Variable declaration, Input/output statement in Pascal, Control structure in Pascal, Array declaration, Sub Program declaration, in Pascal.

Top-down design, Modular design, Stepwise refinement structure Charts, Writing structured programming using Pascal.

CSG 12 : Theoretical Foundations of Computer Science

Mathematical Reasoning, Set theory, Propositional logic, Boolean algebra, Number systems, Relations and functions, Combinatorial circuits, Difference equation, Linear algebra, Coordinate Geometry and coordinate transformation, Probability theory, Graph theory, Automata languages, Turing machines, Algorithm analysis.

CSG 13 : Computer Architecture and Assembly Language Programming

Computer System Organisation: Representation of data, Internal Architecture of CPU instruction sets and addressing, principles of data 1/0, microprocessor systems.

Assembly Language Programming: Detail of an instruction set of a microprocessor and writing simple program.

Computer Input/Output System: Secondary storage, I/O interfaces peripheral devices, Computer communications, Principles and Techniques.

(Prerequisites: CSG 11, CSG12)

CSG 14 : Data Structures

Arrays, Lists, Stacks, Ques, Trees, Graphs, Files and their Applications in Computer Science, Recursion.

Sorting Algorithms: Elementary methods such as Selection sort, Bubble sort, etc., advanced sorting methods such as Tree sort, Quick sort, Heap sort, etc., Study their computation complexity.

(Prerequisites: CSG 11, CSG12)

Second Year Courses

Unit No.	Unit Title
CSG 25	Numerical Methods I
CSG 26	Design of Algorithms
CSG 27	Introduction to Programming in Logic and Expert systems.
CSG 28	Introduction to Database Management Systems

Notes:

1. Each of these units will consist of about 35 hours of lectures plus tutorials and 25 hours of practical.
2. Each unit will be examined by a theory paper and a practical examination.
3. Each theory paper is of two - hour duration containing six questions out of which four should be answered.
4. A practical examination is of three - hour duration
5. For each unit 20% weightage will be given for practicals and 80% for theory.

CSG 25 : Numerical Methods I

Handling of Errors
Fixed point and Floating point arithmetic, implication of finite precision, errors in numerical computation.

Interpolation
Polynomial and interpolations.

Approximations
Uniform approximation, discrete least squares approximation, polynomial approximation, Fourier's approximation, Chebyshev economisation.

Numerical Integration and Differentiation
Interpolatory numerical quadrature, Gaussian quadrature, adaptive integration, Romberg integration, numerical differentiation.

Solutions of Non - Linear Equations
Bisection method, fixed point iteration, Newton's method, Aitken's process, rates of convergence of the various methods.

Algorithms
Algorithms to implement the above methods in a computer using Pascal, C or FORTRAN.

(Prerequisites CSG 11).

CSG 26: Design of Algorithms

Introduction
Definition as a methodology introduced by a mathematician named *Abuja'far Mohammed ibn Musa al Khowarizmi:* characteristics of algorithms; designing, devising and expressing algorithms, use and removal of recursion; validation, analysis, testing and profiling.

Greedy Method
Graph algorithms such as finding spanning tree and single source shortest path etc. Knapsack problem, job sequencing, etc.

Divide and Conquer
Binary search; merge sort; quick sort, etc.

Backtracking
N-queen problem; sum of subsets, graph colouring; Hamiltonian cycle; knapsack problem.

Dynamic Programming
All pairs shortest paths; optimal binary search trees; Knapsack problem; travelling sales person problem.

Branch and Bound
Knapsack problem; travelling sales person problem.

Graph / Tree Algorithm
Breadth-first search in graph (AND-graph, OR-graph etc.); tree traversal on binary, threaded AVL and game trees, handling Btree and disk memory management.

Symbol Table Algorithm
Dynamic tree tables, Hash tables.

(Prerequisites CSG 14)

CSG 27: Introduction to Programming in Logic and Expert Systems

Introduction to Predicate Calculus
Syntax and semantic of predicate calculus, existential and universal quantifiers and variables; first `order predicate calculus; matching of predicate calculus expressions: binding and unification of variables; inference rules: Sound, complete rules: modus ponens, modus tolens, and - elimination, and - introduction, universal instantiation, and resolution etc.

Resolution Procedure
Conversion of predicate calculus expressions into clause forms, horn clauses; resolution refutation and theorem proving.

Prolog Programming
Introduction to prolog as a declarative language as well as a procedural language, syntax of prolog; representation of facts, rules, and queries; predicates, clauses, goals; flow of control; variable binding, unification and backtracing, cut fail, efficiency.

Introduction to Expert Systems
A brief introduction to Artificial Intelligence (A1), its techniques and to expert systems (ES); various definitions of ES, components of ES:" knowledge acquisition and representation; inference mechanism; forward and backward chaining; handling uncertainty; examples of expert systems and expert system shell/tools.

(Prerequisites CSG 12)

CSG 28: Introduction to Database Management System

Introduction
Definition as a centralised storage of database; prevention of redundancy and inconsistency; data independence; data abstraction; data models; data definition and data manipulation language (DDL and DML); database manager, administrator, users; overall system structure: Indexing and hashing.

Entity Model - Relationship Model
Entities, relationships, entity-sets, relationship-sets, attributes. mapping constraints, keys, E-R diagrams, reduction of E-R diagrams to tables, design of an E-R database scheme.

Relational Model
Structure of relational database, relational algebra and calculus;

Normalisation and Relational Database Design
Relational database design and its pitfalls; normalisation (First second, and third) using functional dependencies and multivalued functional dependencies.

Query Processing
SQL, query - by – example, Quell; Query interpretation, equivalence of expressions

Recovery and Concurrency
Transaction recovery, system media recovery, concurrency problems, locking, timestamping.

Security and integrity
General considerations and violations, security specification in SQL; other aspects of security (encryption etc.).

Case Study
Construction and manipulation of database using
dBASE III+, dBASE IV, ACCESS or ORACLE etc. and SQL or alike.

(Prerequisite CSG 14.)

Third Year Courses

Unit No.	Unit Title
CSG 35	Numerical Methods II
CSG 36	Software Engineering
CSG 37	Introduction to Computer Graphics
CSG 38	System Software

Notes:

1. Each if these units except CSG 38 will consist of about 35 hours of lectures plus tutorials and 25 hours practical.
2. CSG 38 will consist of about 45 hours lectures plus tutorials and will be examined solely by a theory paper.
3. CSG 36 will be examined by a theory paper and a project.
4. Each of CSG 35 and CSG 37 will be examined by a theory paper and a practical examination.
5. Theory paper is of two - hour duration containing six questions out of which four should be answered.
6. A practical examination is of three - hour duration.
7. For each unit other than CSG 38, Weightage of 20% for practical / project and 80% for theory will be given, and in the case of CSG 38, 100% will be given for theory paper.

CSG 35: Numerical Methods II

Solving System of Linear Equations
Direct method - Gaussian elimination, direct factorisation methods.

Error Analysis and Norms
Vector norms and matrix norms, condition numbers and error estimates, iterative improvement of solutions.

Iterative Methods
Jacobi method, Gauss-Seidel method, SQR method, criteria for convergence of these methods.

Solving Ordinary Differential Equations
Taylor series method, Euler's method with local and global error analysis, Runge-Kutta method, predictor corrector method, automatic error monitoring change of step size and order, stability.

Algorithms
Algorithms to implement the above methods in a computer using Pascal, C or FORTRAN.

(prerequisite: CSG 25)

CSG 36: Software Engineering

Introduction
Well engineered software, software process and evolution, software reliability, knowledge processing, group working, ergonomics.

Software Specification
Requirements and its evolution; system modelling; context, analysis, model description, real - time system modelling, data modelling; requirements definition and specification; requirements validation and prototyping, formal, algebraic and model - based specification.

Software Design
Top - down design, systems design, design decomposition, quality, design description language; object- oriented design: objects, object classes, inheritance; function - oriented design: data flow diagrams, structure charts; user interface design: objective, metaphors, user guidance; quality assurance.

Programming Techniques
Data abstraction: abstract data typing; portability and reuse; computer-aided software engineering; environment: operating system layer, database layer, object management system,

Software Validation
Program verification and validation: top down, bottom-up testing, scheduling, debugging; testing techniques, verification.

Software Management:
Management activities, structures, programmer productivity; project planning and scheduling; software cost estimation, maintenance, documentation, quality assurance.

Case Study
Application on object-oriented design using Pascal, C++, or Visual BASIC.

(Prerequisite CSG 14)

CSG 37: Introduction to Computer Graphics

Graphics Devices
Input devices: Keyboard, trackball, joystick, mouse, light pen, stylus tablet, sonic tablet, digitizing camera, output devices: raster-scan cathode ray tube (CRT), memory - tube display, plasma display, liquid crystal display, plotters, printers;

Image Storage
Image - 0nly storage, display- memory storage, compressed storage, information storage, run length encoded storage, quad-tree storage.

Scan Conversion
Scan-converting of a point, a straight line a circle, an ellipse, arcs and sectors, a rectangle; region filling; interior- defined and boundary defined regions and filling algorithms; boundary block transfer.

Two - Dimensional graphics
Windows and view ports; clipping algorithms: midpoint algorithms, Sutherland - Hodgman algorithm; introduction to homogeneous coordinates and transformation of object: translation, scaling, rotation, mirror image;

Three - Dimensional Graphics
Parallel and perspective projections; clipping algorithms, transformation of objects; hidden line and hidden surface removal

Graphics Models
Mathematical models for two - dimensional and three dimensional curves and curved - surfaces (Bezier curves and B- splines etc.)

GKS
introduction to graphics kernel systems (GKS) and the counter parts in Turbo Pascal / C/ BASIC, Visual BASIC, QBASIC

(Prerequisite CSG 14)

CSG 38: System Software

Introduction
Definition and history of operating systems; structure of an operating system; parallel and distributed computation.

Process Management
Concept of process, semaphores, concurrent process and programming, deadlock: prevention avoidance, detection, recovery and postponements

Storage Management
Real and virtual storage management: basic concepts contiguous and non - contiguous storage allocation: fixed and variable partition multiprogramming, storage swapping, paging/ segmentation systems;

Processor Management
Job and processor scheduling; distributed computation view.

Auxiliary Storage Management
Disk performance optimisation; file and database systems

Performance
performance coprocessors , reduced instruction set computing (RISC), data flow; analytic modelling: queuing theory, Markov process

Networks and Security
Distributed computing - open systems interconnection view, operating systems security.

Compilers
Introduction to compilers, lexical analysis, syntax analysis and code generation.

Case Study;
MS - DOS, PC-DOS, OS/2, UNIX.

(Prerequisite: CSG 13, CSG 26)